ChinaXiv合作期刊

一种结合遗传算法的工控协议模糊测试方法 *

张冠宇 [1,2,3]，尚文利 [1,3,4,5†]，张博文 [1,3,4,5]，陈春雨 [1,3,4,5]，刘周斌 [6]，张　锐 [2]

(1. 中科院网络化控制系统重点实验室，沈阳 110016; 2. 沈阳建筑大学 信息与控制工程学院，沈阳 110168; 3. 中国科学院沈阳自动化研究所，沈阳 110016; 4. 中国科学院机器人与智能制造创新研究院，沈阳 110169; 5. 中国科学院大学，北京 1000396; 6. 国网浙江省电力有限公司电力科学研究院，杭州 310014)

摘　要：模糊测试(fuzzy test)在工控协议的漏洞挖掘中有很好的适用性，但传统的模糊测试存在着用例的生成工作量大、失效率高等弊端。为了解决这些问题，设计了一个结合遗传算法(genetic algorithm)与模糊测试的工控协议模糊测试器 GA-fuzzer，并引入基于维度变换的用例空间模型和危险点的概念。在 GA-fuzzer 中，构造了更有效的动态适应度函数，同时设计了动态变异算子和交叉算子，优化测试用例。在相同实验环境下，分别采用开源模糊测试方法 Peach 以及 GA-Fuzzer 对目标进行测试，结果显示 GA-fuzzer 可有效的改善传统遗传算法的过早收敛问题，且与 Peach 相比，达到相同的测试预期所使用的用例数量降低 27.20%，测试时间降低 34.82%。

关键词：工控协议测试；遗传算法；模糊测试；漏洞挖掘

中图分类号：TP29　　doi: 10.19734/j.issn.1001-3695.2020.03.0048

Fuzzy test method for industrial control protocol combining genetic algorithm

Zhang Guanyu[1,2,3], Shang Wenli[1,3,4,5†], Zhang Bowen[1,3,4,5], Chen Chunyu[1,3,4,5], Liu Zhoubin[6], Zhang Rui[2]

(1. *Key Laboratory of Networked Control Systems*, *Chinese Academy of Sciences*, *Shenyang* 110016, *China*; 2. *School of Information & Control Engineering*, *Shenyang Jianzhu University*, *Shenyang* 110168, *China*; 3. *Shenyang Institute of Automation*, *Chinese Academy of Sciences*, *Shenyang* 110016, *China*; 4. *Institutes for Robotics & Intelligent Manufacturing*, *Chinese Academy of Sciences*, *Shenyang* 110169, *China*; 5. *University of Chinese Academy of Sciences*, *Beijing* 1000396, *China*; 6. *Electric Power Research Institute of State Grid Zhejiang Electric Power Co, Ltd. Hangzhou* 310014, *China*)

Abstract: Fuzzy Test has good applicability in the exploitation of vulnerabilities in industrial control protocols. However, the traditional fuzzy test has the disadvantages of large test workload and a high failure rate. In order to solve these problems, it design an industrial control protocol fuzzy tester GA-fuzzer which combines genetic algorithm and fuzzy test. and propose the concepts of dangerous points and case space model based on dimensional transformation. In GA-fuzzer, it constructed a more efficient dynamic fitness function, and design dynamic mutation and crossover operators to optimize test cases. In the same experimental environment, it used open source fuzzy test method Peach and GA-Fuzzer to test the target. The results show that GA-fuzzer can effectively improve the premature convergence problem of traditional genetic algorithm, and compared to Peach, the number of cases used to achieve the same test expectation was reduced by 27.20% and the test time was reduced by 34.82%.

Key words: industrial control protocol test; genetic algorithm; fuzzy test; vulnerability mining

chinaXiv:202009.00100v1

0　引言

工业控制系统(industry control system，ICS) 是工业互联网系统的“控制大脑”，是由监控数据采集系统(SCADA)、分布式控制系统(DCS)、过程控制系统(PCS)、可编程控制器(PLC)以及远程终端单元(RTU)等组成[1]的各控制系统的总称，广泛应用于石油化工、机械制造、电力、水利设施、金融服务、商业设施、交通系统和通信等支撑现代社会的重要行业以及部门[2]。从广义的角度来看，凡是涉及到工业控制设备的应用场合，都离不开工业控制系统的支持，它是整个社会赖以生存的基础条件之一。

传统的工业控制系统在最初的设计中更加注重功能性，对安全问题的考虑有所不足，而且多属于专用系统。这类系统内部封闭，较少与外部联系，导致许多关键基础设施的控制系统，很少有强力的防御攻击的保护措施[3]。而从过去到现在的一系列工控安全事件都表明，工业控制系统的安全性是一个需要不断完善和加强的研究工作，这就要求工控系统的安全防范需要从可被攻击的可各方面着手，并建立相应的安全保障措施[4,5]。

通信协议作为工控系统的数据信息采集、控制指令的传递载体，其安全性至关重要。传统的工控协议在安全方面考虑欠缺，比如对信息加密不足、无协议认证、部分协议的文档公开等，这都使得采用这类协议的系统存在较大的安全问题，而工控协议的安全漏洞是较为容易被攻击者利用的突破口之一[6]。这种攻击成本低，收益高，攻击者只要掌握了协议规约，便可以伪造针对协议中漏洞的数据报文，进而对整个

收稿日期：2020-03-13；**修回日期：**2020-04-16　**基金项目：**国家重点研发计划项目(2018YFB2004200)；国家自然科学基金项目(61773368)；2019 年工业互联网创新发展工程—工业企业网络安全综合防护平台项目；国家电网公司科技项目(52110418001B)

作者简介：张冠宇(1991-)，男，河南省周口人，硕士研究生，主要研究方向为工业控制系统信息安全、漏洞挖掘等；尚文利(1974-)，男(通信作者)，黑龙江北安人，研究员，博导，博士，主要研究方向为工业控制系统信息安全、计算机智能与机器学习等(shangwl@sia.cn)；张博文(1994-)，男，辽宁沈阳人，助理研究员，硕士，主要研究方向为信息安全、嵌入式系统、人工智能等；陈春雨(1992-)，男，黑龙江哈尔滨人，助理研究员，硕士，主要研究方向为信息安全、人工智能等；张锐(1978-)，女，辽宁沈阳人，副教授，博士，主要研究方向为神经网络、深度学习等；刘周斌(1972-)，男，浙江嘉兴人，高工，硕士，主要研究方向为电力信息技术、网络安全等.

ChinaXiv合作期刊

工控系统造成危害。因此对于工控协议的漏洞挖掘，是保障工控系统安全不可或缺的一部分。

针对通信协议的安全研究，国外的发展研究起步早、发展快、技术新，并且逐渐趋于成熟，开发出多种测试工具应用于实际。国内的工控协议模糊测试经过近几年的发展也逐渐发展完善。Artemios G. Voyiatzis 等人[7]设计了一个模糊测试器，用于对 Modbus/TCP 可能存在的安全漏洞进行挖掘，它在测试中加入一个侦测阶段，以帮助测试器有更好的预测能力，及时调整攻击向量，这样大大降低了模糊测试的随机性；在文献[8]中提出了利用遗传算法的变异算子来处理多维输入，得到了目标应用中输入元素与不安全函数之间的影响关系，通过对遗传算子的操作来触发可以漏洞；Banks G 等人在文献[9]中，针对模糊测试在协议安全测试中的应用受限的问题，构造了一个适用于网络协议的模糊测试器 SNOOZE，它允许测试者描述协议的各状态及各状态所需的消息报文，有利于对特定的协议状态进行针对性测试；在文献[10]中，Kang J 等人提出了一种结合黑盒与白盒测试的模糊测试系统，该系统基于弱点分析方法准确的检测安全弱点，挖掘被测目标的潜在威胁；在国内，赖英旭等人[11]在模糊测试中，通过引入变异因子这一概念，也就是工控系统漏洞特征，来生成 Modbus/TCP 协议测试所需的用例数据，并通过旁路监听来确定用例的有效性，最终证明模糊测试应用到工控协议的测试上是行之有效的；在文献[12]中，涂玲等人为了解决用例覆盖率低、测试结果无法准确评价的问题，提出了协议变形和动态特征并行混合的测试用例生成方法，该方法在提高用例覆盖率的同时，降低了结果和漏洞误报率；张亚丰等人[13]提出基于状态的工控协议模糊测试技术，设计了一个基于协议状态机的测试序列生成算法，解决了以往协议测试中未考虑协议交互状态和测试方式、检测手段受限的问题，最终实现了较高的漏洞命中率和目标覆盖率。

目前模糊测试在工控协议的漏洞挖掘中的应用研究，主要是寻找更有效的用例生成手段以及降低模糊测试的随机性，从而达到降低用例数量，提高测试效率的目的。本文结合目前的工控协议漏洞挖掘的研究现状，提出了一种针对工控协议的漏洞挖掘的模糊测试方法。该方法结合遗传算法，设计了动态适应度函数来控制用例的生成。同时为了评价用例的优劣，提出用例空间和危险点的概念，来计算用例之间的相似度和重要度特征，与动态适应度函数结合，可以有效降低测试用例失效率，减少测试用例数量。

1 相关理论

1.1 Modbus 通信协议

Modbus 通信协议自 1979 年由施耐德电气发表，由于部署简单、易于维护等优点，逐渐成为工业控制系统中最常见的数据通信方式。它的实质是一种主从式的通信协议，定义了一个控制请求设备如何访问其他服务设备的过程[14]。

根据传输数据的格式、物理接口等条件的不同，Modbus 通信协议可以分为适用于串行链路的 Modbus RTU、Modbus ASCII，以及通过 TCP/IP 传输的 Modbus/TCP 等多种模式。为了方便在不同模式下的数据传输，Modbus 通信协议定义了一个与基础通信层无关的数据应用单元(Protocol Data Unit，PDU)，在不同的传输模式下，只需要在其首位加上相应的附加域，便可以正常的传输数据信息。

图 1 是在 TCP 模式下的数据帧结构，其中应用数据单元(application data unit，ADU)包括六部分：事务处理标志符用于给每次的通信事件编号，请求与响应相同；协议标志符表明该数据帧所传输的数据所遵从的协议类型，Modbus 的标志是 0x0000；长度域存储包括单元标志符以及 PDU 段在内的字节个数；单元标志符是系统内路由，即串行链路的设备地址编码；最后是 PDU，用于存储该次通信的数据信息。在这种模式下，基于串行协议的 PDU，在其头部依次增加事务处理标志符、协议标志符、长度信息和单元标志符，再封装上 TCP 的协议报文头部，便可以实现 Modbus 在以太网上的传输，本文的测试均在该种传输模式下完成。

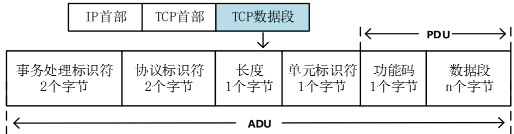


图 1 Modbus TCP/IP 数据帧结构

Fig. 1 Modbus TCP/IP data frame structure

1.2 模糊测试

模糊测试(fuzzy test)最早是由 Barton Miller 教授在 1989 年提出，它的基本思想是构造大量的畸形数据作为测试目标的输入，然后观测目标的响应结果来分析漏洞。由于不需要了解目标的内部结构，只要掌握输入的数据格式就可以构造用例发送给目标，这使得模糊测试有着极强的适应性和可移植性。

模糊测试根据不同的测试目标、测试用例生成方法，测试的过程可能会有所差别，这取决于测试目标的特性、研究者的技能以及测试用例的数据格式等。但是无论采用何种方法，模糊测试都会包括六个基本步骤：

a) 确定测试目前信息，包括目标类型、历史漏洞信息等。

b) 确定测试数据输入量、数据编码格式等所有需要输入的数据类型。

c) 根据输入信息，设计用例生成方法，构造测试所需的用例集合。

d) 依据目标的数据传输规约，向目标发送测试用例。

e) 设置监视器，检测测试运行结果。依据目标的数据传输规约，向目标发送测试用例。

f) 根据最终结果，判定是否存在可被利用的漏洞。

在工业控制系统中，设备之间的信息安全传输依靠协议的稳定，它关系着工控系统的安全。但工控系统的工作环境复杂多样，又导致测试所需的工作量较大。而模糊测试的高适用性和可移植性，可以很好的解决这一问题。在文献[15]中表明，模糊测试经过针对性的设计，在工控系统的安全测试中有较好的适用性，能够挖掘系统通信协议中的潜在安全问题。

2 GA-fuzzer 的设计与实现

本文利用模糊测试方法，结合 Modbus 协议的特征对其进行测试。在传统的模糊测试中，生成的测试用例失效率高，同样在对工控系统协议进行漏洞挖掘时，也会出现大量无效报文，这就导致测试的效率低下[16]。为了解决模糊测试应用到工控协议漏洞挖掘时测试用例是效率高的问题，同时让这个过程自动化且高效，本文将遗传算法引入到模糊测试中，在原有的遗传算法基础上，作出策略调整，用于测试用例的生成以及优化。

遗传算法是最早的全局寻优算法之一，它通过模仿生物界的自然选择理论，在每一代种群内搜索最优个体，最终通过种群在迭代中选择、交叉以及变异等操作，找到最优解[20]。虽然遗传算法能够很好的解决全局寻优的问题，但传统遗传算法会在种群迭代过程中逐渐收敛，导致生成的用例相似度逐渐增大，显然是不能直接用于测试用例的生成，因此需要对遗传算法的遗传策略进行相应的改进调整。

2.1 相关改进及优化

对现有的遗传算法的改进及优化，主要针对的是传统遗传算法的过早收敛，以及模糊测试中用例失效率高、测试效

chinaXiv:202009.00100v1

ChinaXiv合作期刊

率低的问题等。主要的工作有以下几个方面：

测试用例编码方式：工控协议的通信报文，其本质也是一系列符号的有序序列。因此可以用一维向量表示一个测试报文序列，其结构为 $case=[c_1\ \ c_2\ \ c_3\ \ \dots\ \ c_n]$，其中 c_n 为该测试报文相应字段的数据，其构成包括整数、字符、分隔符以及格式化字符等。那么遗传算法中的一代种群便可以用矩阵 T 表示，如式(1)。初始种群内的个体，编码采用二进制，对应的协议中每个字节由 8 位二进制来表示。经过测试服务器的字节流转换便可以向 Modbus 客户端发送用例。

$$T=\begin{bmatrix} c_{11} & c_{12} & c_{13} & \dots & c_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{m1} & c_{m2} & c_{m3} & \dots & c_{mn} \end{bmatrix} \tag{1}$$

选择算子：选择环节是本文的一个重要部分。遗传算法的适应度函数将指导整个算法的走向。因此在进行 Modbus 协议的模糊测试时，合适的适应度函数显得尤为重要。在设计适应度函数之前，首先提出两个重要概念：用例空间和危险点。

Modbus/TCP 的每条数据报文的相应字段有固定的功能，且存在一定的范围，例如从设备地址码的范围是 0~255。因此将报文的各字段通过归一化处理，然后映射到对应维度的空间内，便可以将每条测试用例与用例空间内的点一一对应。对任意的一个测试用例 $case$ 来说，该用例的第 n 个字段对应的坐标值 v_n 的计算公式为式(2)，式中 c_{n-max}、c_{n-mix} 分别是该种群内个体的第 n 个字段的对应的最大值和最小值。

$$v_n=\frac{c_n-c_{n-mix}}{c_{n-\max}-c_{n-mix}} \tag{2}$$

那么该条测试用例映射到用例空间即可表示为 $case=\{v_1\ \ v_2\ \ v_3\ \ \dots\ \ v_n\}$。将所有的用例都映射到用例空间内后，每条用例都有自己唯一的空间坐标与之唯一对应，此时便可以引入各个空间点之间的欧氏距离来表示两个用例 i 和 j 之间的相似程度，可用式(3)描述为，式中 m 的值是 i 和 j 中维度较小者。

$$d_{ij}=\left(\sum_{1}^{m}\left(i_{v_m}{}^2-j_{v_m}{}^2\right)\right)^{\frac{1}{2}} \tag{3}$$

这里对高维的测试用例进行降维处理，虽然这样会损失一定的数据信息，但是这些数据多是 Modbus 协议报文的数据存储字段，对用例的相似度以及实验结果的负面影响可以忽略。

在用例生成后发送给测试服务器时，根据响应状态判断是否可以作为危险点。用例的响应状态一共有四种：正常响应，测试用例报文功能正常，服务端可以正常响应；异常响应，测试用例报文字段异常，但服务端可以正常识别该异常并响应；请求超时，测试用例出现未知异常，服务端不能正常响应；其他，测试用例出现未知异常，导致服务端关闭通信、拒绝访问等异常。

上述四种状态中，若响应状态为正常响应或者异常响应，那么测试用例属于 Modbus 协议的正常报文，都可以被测试服务端识别；如果为请求超时，说明测试用例的字段信息不符合 Modbus 协议规约，用例不能被测试服务端识别，属于无效用例；当响应出现其他状态，测试服务端出现严重错误，例如不对后续的用例响应、处于关闭或者拒绝服务等异常状态，该用例有大概率引发测试服务器的异常，将引发该状态的用例标记成危险点。

将危险点对应的用例坐标标记在用例空间内，在种群进行遗传算法迭代时，计算个体与危险点之间的距离便可以给出该点的重要度。当空间内有 q 个危险点时，首先计算每个用例点与各个危险点的距离，比较后将其归入离其最近的危险点对应的一类内，此时种群将被划分为 q 个子种群，且每一个子种群与危险点一一对应。

用例平均相似度：尽可能提高用例的覆盖率是测试用例生成过程中必须要考虑的一点。用例平均相似度可以作为反映用例多样性的一个指标，通常采用个体间的平均距离进行描述。但每个个体都需要计算本身与其他所有个体的距离，导致该方法出现大量的重复计算，效率较低。本文设计了一个新的计算方法，可以准确地反映出种群内个体的分布情况，并且降低计算量。

首先对整个种群内所有个体的四个字段进行求和并取均值，得到中心用例 $\overline{case}=[\bar{i}\ \ \bar{u}\ \ \bar{f}\ \ \bar{d}]$，其中各字段的计算公式如式(4)所示，其中 m 是当前种群内个体数量，x 为用例的四个字段。

$$\bar{x}=\frac{1}{m}\cdot\sum_{i=1}^{m}x_i \tag{4}$$

则每条用例的平均相似度可以用式(3)求得的其与中心用例的距离来表示。这种方法计算用例平均相似度的时间复杂度为 $O(n)$，相较于一般方法 $O(n\lg n)$ 的时间复杂度，在 n 较大时会有显著的效率提升。

以用例空间、危险点以及用例平均相似度为基础，本文提出了一种新的动态适应度函数来描述用例的优劣，其计算方法受用例空间内是否有危险点影响：

a) 在种群用例空间内没有危险点，即 $q=0$ 时，相似度函数是选择操作的主导因素，第 k 条测试用例的平均相似度，即此时的个体适应度函数计算公式如式(5)：

$$f(x)=1-\frac{d_{k\text{-}\overline{case}}}{d_{\max}},\quad k=1,2\dots \tag{5}$$

b) 当用例空间内出现危险点，即 $q>0$ 时，应该首先将种群划分成 q 个子种群，每个个体将其归于与之距离最近的危险点。然后在各子种群之间独立进行选择操作，此时采用重要度函数作为适应度函数，也就是个体与该子种群所属的危险点之间的距离。第 p 个危险点对应的子种群中第 t 个个体的适应度函数如式(6)，式中 d_{pt} 是个体与该子种群所属的危险点之间的距离，$d_{\max-p}$ 是其中的最大者。

$$f_p(x)=\frac{d_{t-p}}{d_{\max-p}} \tag{6}$$

然而由于危险点并不一定是可利用的漏洞，因此为了提高测试效率，给每个危险点设置了 20 代的存活周期。当达到最大存活周期后，记录危险点的测试信息后将其消除。

交叉与变异算子：交叉和变异算子(p_c、p_m)是遗传算法中保持种群多样性的关键算子。由于种群在迭代时不能有过高的个体相似度，因此交叉、变异的概率对种群的多样性有较大影响。文献[20]中主提出的自适应交叉、变异算子设计，可以有效的根据种群状态实时的调整变异以及交叉概率，本文将用例构成的种群状态与该公式结合，构造出了适应于 Modbus 模糊测试的自适应概率函数如式(7)：

$$p=\begin{cases} P\cdot\left(1-\dfrac{d_{ave}}{d_{\max}}\right)^{-1} & \dfrac{d_{ave}}{d_{\max}}>a,\dfrac{d_{\min}}{d_{\max}}>b,p<1-b \\ P & \text{其他} \end{cases} \tag{7}$$

其中 d_{ave} 为当前(子)种群内所有个体间的平均距离，$d_{\max}$、$d_{\min}$ 分别是最大距离与最小距离。参数 a、b 用于判断当前种群是否相似度过高，文中取值经验值 $a=0.72$，$b=0.25$[18]。这样构造的交叉、变异概率，其大小随种群的状态发生改变：当 d_{ave} 较小，即种群内的测试用例之间相似度较高时，此时算得的概率 P 值较大，进而使得当前遗传迭代的种群个体有更高的变异概率，提高多样性；当 d_{ave} 较大时，则与之相对反。

基于前文所述的理论方法，设计的遗传算法流程如图 2 所示，其中的主要步骤有以下几个部分：

chinaXiv:202009.00100v1

ChinaXiv合作期刊

chinaXiv:202009.00100v1

a) 构造初始用例种群用于用例的生成与优化，输入到算法中；

b) 根据当前种群用例空间状态、危险点个数选择适应度函数，若不存在危险点，转到步骤 c) ，若存在危险点，转到步骤 d) ；

c) 选择用例平均相似度作为适应度函数，并进行用例间相似性检验，转到步骤 e) ；

d) 首先根据用例所属危险点将种群划分为若干子种群，选择个体重要度作为适应度函数，转到步骤 e) ；

e) 根据个体适应度在种群内或者子种群内执行遗传相关操作；

f) 判断是否满足终止条件：若不满足，根据更新后的危险点信息，在用例空间内标记新增危险点，返回步骤 b) ；若满足终止条件，结束算法；

g) 算法终止，输出测试日志。

相较于传统遗传算法，主要加入了动态适应度函数，通过监测用例种群内是否存在危险点用例选择不同的遗传算法适应度函数，避免了传统遗传算法在应用的用例生成时，一旦收敛之后，用例出现较高的相似性，导致测试用例失效率迅速升高的问题。同时在加入的动态变异及选择概率，可以根据种群状态调整种群内用例的多样性，尽可能的在提高测试命中率的同时提高用例的覆盖率。

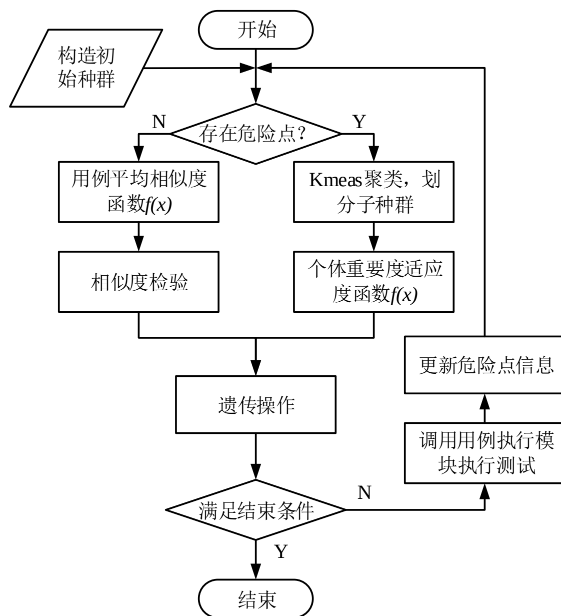


图 2　基于遗传算法的测试用例生成算法

Fig. 2　Case generation algorithm based on genetic algorithm

2.2　GA-fuzzer 设计

结合改进的遗传算法和模糊测试，本文设计的 GA-fuzzer 如图 3 所示。

a) 协议分析：分析 Modbus 协议报文特征，对协议报文的各字段数据类型、功能特性以及取值范围作标记，同时捕获正常的通信报文，构造出较高质量的初始测试种群。

b) 用例生成：将构造好的初始种群发送给用例生成单元，该环节的功能包括危险点标记、计算各动态适应度函数、种群个体的遗传操作、更新用例空间等。

c) 仿真测试：在种群优化完毕后，该代所有的个体经过 Modbus 模拟服务器发送给测试目标进行测试。

d) 异常检测：将每一条测试用例的运行结果记录到测试日志，并确认是否存在危险点，以此更新危险点信息、适应度函数、变异概率和交叉概率。

e) 结果分析：分析实验最终结果，确定是否存在可被利用的安全漏洞。

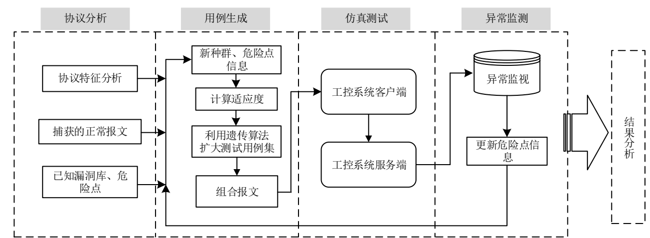


图 3　GA-fuzzer 结构设计

Fig. 3　Structural design of GA-fuzzer

3　实验及结果分析

为了验证所提方法的有效性以及测试其性能表现，分别采用开源模糊测试方法 Peach 以及本文所提方法对目标进行测试。Peach 是由 Michael Eddington 等人在 2004 年发布的一个开源模糊测试工具，其测试对象包括文件格式、通信协议、API 等。通过对 Peach Pit 文件的配置，包括数据类型、变异策略、数据长度等，可以实现对工控协议模糊测试[19]。

实验环境：实验硬件平台为 Intel(R) Core™ I5-3470 CPU @ 3.20GHz; 3.79GB RAM; Windows 10; 开发语言为 Python3.6; 测试采用 Modbus 通信仿真软件。首先使用 Modbus Poll 与 Modbus Slave 建立正常的数据通信，使用 Wireshark 抓包工具获取正常的通信报文，从中选取具有代表性的数据报文进行数据特征的分析，并以此为基础构造初始种群，并将其作为各测试方法的输入，对目标进行测试。

两组实验中的主要相同参数设置为：种群规模 $size=100$；交叉标准概率 $p_c=0.76$；变异标准概率 $p_m=0.03$。

实验终止条件为挖掘出漏洞，首先为了测试前文中提出的用例优化方法的性能，对两种方法所生成的用例以种群为单位统计每一代的用例平均相似度，如图 4 所示，其中用例平均相似度的计算方法如式(8)所示。

$$s=1-\frac{1}{n\cdot d_{\max}}\cdot\sum_{i=1}^{n}d_{i-case} \tag{8}$$

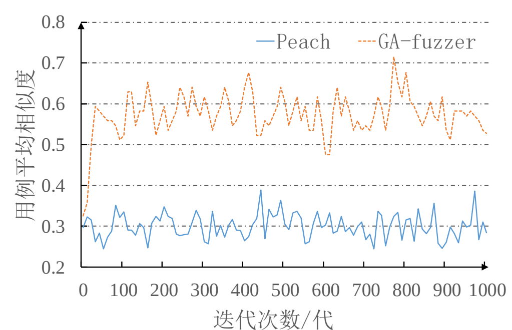


图 4　用例平均相似度变化趋势

Fig. 4　Trend of Case Average Similarity

对结果进行分析后可知，在 Peach 测试中，变异概率在测试前由配置文件给定后，并不能通过判断用例生成的状态进行调整，虽然用例的平均相似度较低，但针对性较弱，用例的生成趋向于随机。而 GA-fuzzer 的用例种群的状态动态调整，当危险点增多时，由于遗传算法的收敛特性，会导致生成的用例逐渐收敛于所属危险点，此时对变异概率进行适当的调整，可提高危险点附近的漏洞命中率。

另外，在结果中有部分种群的用例相似度超过了 0.7，例如 770 代左右的种群，经过对原始数据分析，该种情况下为种群内存在危险点，且趋于收敛。根据前文提到的动态适应度函数，此时的用例生成趋向于危险点，提高用例的漏洞命中概率，因此出现用例间相似度较高。

在两组实验在相同环境下分别进行多次测试后，对生成的用例进行统计，主要测试数据的平均值如表 1 所示。

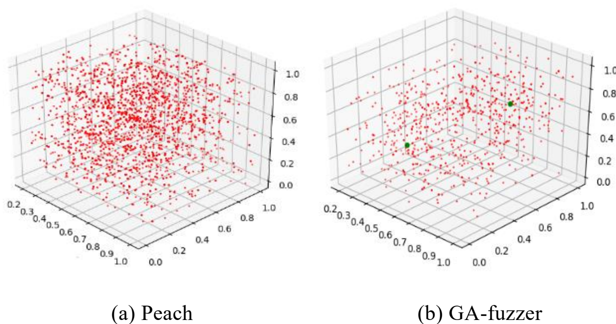
ChinaXiv合作期刊

表 1 实验结果数据对比

Tab. 1 Comparison of experimental results

生成方法	用例数量/个	相似度	时长/h	漏洞数/个
GA-fuzzer	98561	0.4381	14.60	1.00
Peach	135384	0.3354	22.40	0.80

同时在三组测试中选取其中具有代表性的连续 100 代种群作为示例数据，归一化后抽象到三维用例空间内，可以得到各测试方法所生成的用例在用例空间内的分布情况，如图 5 所示。



(a) Peach (b) GA-fuzzer

图 5 各方法生成的用例在用例空间内的分布

Fig. 5 Distribution of cases of each method in the case space

从各测试方法的测试用例在用例空间内的分布状态来看，开源模糊测试框架所产生的用例，在用例空间内的分布较为均匀，如图 5(a)。这是由于其用例的生成没有较好的控制策略，更多的是趋向于更高的覆盖率。该方法虽然能够保持个体间有一定的差异性，但是针对性不理想，不能快速的定位漏洞，这导致测试所需的时间较长，且需要大量的测试用例；图 5(b)是采用 GA-fuzzer 的模糊测试方法所生成的测试用例。示例数据中，存在两个危险点，根据所提方法的设计，在用例空间内存在危险点时，种群内个体首先会根据距其较近的危险点进行聚类，然后在所在类的内部各自进行遗传收敛，这样可以同时对多个可疑漏洞进行针对测试，提高测试效率；在空间内没有危险点或者所有危险点的存活周期结束时，用例的生成倾向于高覆盖、低相似的趋势，保证测试具有较高的覆盖率。通过对空间内是否存在危险点来对遗传算法的适应度函数、遗传算子进行动态更新，可以同时具有较高的测试覆盖率和很好的针对性，有效的降低测试用例的失效率。表 1 中的数据也表明，在相同的实验环境中，成功完成测试的情况时，与 Peach 模糊测试工具相比，本文所提的 GA-fuzzer 模糊测试器在时间效率上有一定的优势，可以将测试时长缩短约 34.82%，同时测试所需的测试用例数量也降低了约 27.20%。

4 结束语

本文将遗传算法应用到工控协议的模糊测试中，设计了一个利用用例空间来动态调整遗传算法适应度函数、交叉以及变异概率的模糊测试器，为模糊测试在工控协议漏洞挖掘上的应用提供了一个新的思路和方法：用例空间和危险点的概念在一定程度上弥补了模糊测试的用例生成随机性强的不足，同时也解决了传统遗传算法过早收敛的问题；动态适应度函数的加入，使得遗传算法的遗传操作可以根据种群内个体的状态实时调整，生成的测试用例更加具有针对性，降低测试用例失效率提高测试效率。下一步工作将针对用例空间模型做深一步的研究，比如更加细致的处理因降维造成的数据损失，优化动态适应度函数的调整策略，进一步提升生成的测试用例质量。

参考文献：

[1] 闫腾飞，尚文利，赵剑明，等. 基于遗传算法优化的 OCSVM 双轮廓模型异常检测算法 [J/OL]. 计算机应用研究，TP. 20180811: 1-3. (Yan Tengfei, Shang Wenli, Zhao Jianming, *et al.* Anomaly detection algorithm based on OCSVM double contour model of genetic algorithm optimization for industrial control system [J/OL] , Application Research of Computers, TP. 20180811: 1-3.)

[2] 安高峰，朱长明，雷晓锋，等. 我国工业控制系统信息安全政策和标准体系架构研究 [J]. 信息安全研究，2018, 4 (10): 959-964. (An Gaofeng, Zhu Changming, Lei Xiaofeng, *et al.* Information Security Policy and Standard System of Industrial Control in China [J]. Research on Information Security, 2018, 4 (10): 959-964.)

[3] 万明，尚文利，曾鹏，等. 基于功能码深度检测的 Modbus/TCP 通信访问控制方法 [J]. 信息与控制，2016, 45 (02): 248-256. (Wan Ming Shang Wenli Zeng Peng, *et al.* Modbus/TCP Communication Control Method Based on Deep Function Code Inspection [J] , Information and Control, 2016, 45 (02): 248-256.)

[4] 熊琦，彭勇，伊胜伟，等. 工控网络协议 Fuzzing 测试技术研究综述 [J]. 小型微型计算机系统，2015, 36 (3): 497-502. (Xiong Qi, Peng Yong, Yi Shengwei, *et al.* Survey on the fuzzing technology in industrial network protocols [J]. Journal of Chinese Computer Systems, 2015, 36 (3): 497-502.)

[5] Edmonds J. Security Analysis of Multilayer Protocols in SCADA Networks [D]. Department of Computer Science, University of Tulsa, Tulsa, Oklahoma, 2006.

[6] Huitsing P, Chandia R, Papa M, *et al.* Attack taxonomies for the Modbus protocols [J]. International Journal of Critical Infrastructure Protection, 2008, 1 (none): 37-44.

[7] Voyiatzis A G, Katsigiannis K, Koubias S, *et al.* A Modbus/TCP Fuzzer for test internetworked industrial systems [C]. 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA) . IEEE, 2015.

[8] Wu Zhiyong, Atwood J W, Zhu Xueyong. A New Fuzzing Technique for Software Vulnerability Mining [J]. Journal of Communication and Computer, 2011 (2): 88-95.

[9] Banks G, Cova M, Felmetsger V, *et al.* SNOOZE: Toward a Stateful Network protocol fuzzer [C] Information Security, 9th International Conference, ISC 2006, Samos Island, Greece, August 30-September 2, 2006, Proceedings. DBLP, 2006.

[10] Kang J, Park J H. A secure-coding and vulnerability check system based on smart-fuzzing and exploit [J]. Neurocomputing, 2017, 256 (20): 23-34.

[11] 赖英旭，杨凯翔，刘静，等. 基于模糊测试的工控网络协议漏洞挖掘方法 [J/OL]. 计算机集成制造系统: TP20180511, 1-22. (Lai Yingxu, Yang Kaixiang, Liu Jing, *et al.* A Vulnerability Mining Method for Industrial Control Network Protocol Based on Fuzz Test [J/OL]. Computer Integrated Manufacturing Systems: TP 20180511, 1-22.)

[12] 涂玲，马跃，程诚，等. 基于协议混合变形的 Web 安全模糊测试与效用评估方法 [J]. 计算机科学，2017, 44 (05): 141-145. (Tu Ling, Ma Yue, Cheng Cheng, *et al.* Hybrid Protocol Deformation Based Web Security Fuzzy Test and Utility Evaluation Approach [J]. Computer Science, 2017, 44 (05): 141-145.)

[13] 张亚丰，洪征，吴礼发，等. 基于状态的工控协议 Fuzzy Test 测试技术 [J]. 计算机科学，2017, 44 (05): 132-140. (Zhang Yafeng, Hong Zheng, Wu Lifa, *et al.* Form-syntax based Fuzzing method for industrial control protocols [J]. Computer Science, 2017, 44 (05): 132-140.)

[14] 程杨，刘学平，占涛. 一种基于 MODBUS 协议的工业控制系统设计 [J]. 机械设计与制造，2011 (01): 1-3. (Cheng Yang, Liu Xueping, Zhang Tao, Design of an Industrial Control System Based on MODBUS Protocol [J]. Machinery Design & Manufacture, 2011 (01): 1-3.)

chinaXiv:202009.00100v1

ChinaXiv合作期刊

[15] 崔欣，温彦龙．工业控制系统模糊测试研究与应用 [J]．信息安全与通信保密，2018 (9): 73-78. (Cui Xin, Wen Yanlong, Analysis and application of fuzzy test protocol for industrial control systems [J]. China Information Security, 2018 (9): 73-78.)

[16] 张亚丰，洪征，吴礼发，等．基于范式语法的工控协议 Fuzzy Test 测试技术 [J]．计算机应用研究, 2016, 33 (8): 2433-2439. (Zhang Yafeng, Hong Zheng, Wu Lifa, *et al.* Form-syntax based Fuzzing method for industrial control protocols [J]. Application Research of Computers, 2016, 33 (8): 2433-2439.)

[17] 于海斌，曾鹏，尚文利等．CN 105721230，一种面向 Modbus 协议的模糊测试方法：中国 [P/OL]. 2016. 06. 29. (Yu Haibin, Zeng Peng, Shang Wenli, *et al.* CN 105721230 A fuzzy test method for Modbus protocol: China [P/OL]. 2016. 06. 29.)

[18] 李柱．基于自适应遗传算法的软件测试用例自动生成 [J]．计算机系统应用，2016, 25 (01): 192-196. (Li Zhu, Automatic Test-Case Generation Based on Adaptive Genetic Algorithm [J]. Application of Computer System, 2016, 25 (01): 192-19)

[19] 伊胜伟，张翀斌，谢丰，等．基于 Peach 的工业控制网络协议安全分析 [J]．清华大学学报：自然科学版, 2017, 57 (1): 50-54. (Yi Shengwei, Zhang Chongbin, Xie Feng, *et al.* Security analysis of industrial control network protocols based on Peach [J]. Journal of Tsinghua University: Science &Technology, 2017, 57 (1): 50-54.

chinaXiv:202009.00100v1